

P24215.P07

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant : Toshimasa TAKAKI et al.

Appln No. : 10/662,449 Group Art Unit: Unknown

Filed : September 16, 2003 Examiner: Unknown

For : DATA PROCESSING APPARATUS, DATA PROCESSING METHOD, AND
DATA PROCESSING PROGRAM

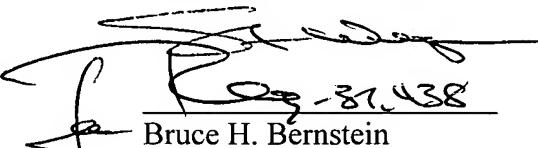
**SUPPLEMENTAL CLAIM OF PRIORITY
SUBMITTING CERTIFIED COPY**

Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Sir:

Further to the Claim of Priority filed September 16, 2003 and as required by 37 C.F.R. 1.55,
Applicant hereby submits a certified copy of the application upon which the right of priority is
granted pursuant to 35 U.S.C. §119, i.e., of Japanese Application Nos. 2002-284164, filed
September 27, 2002 and 2003-305258, filed August 28, 2003.

Respectfully submitted,
Toshimasa TAKAKI et al.


Bruce H. Bernstein
Reg. No. 29,027

December 9, 2003
GREENBLUM & BERNSTEIN, P.L.C.
1950 Roland Clarke Place
Reston, VA 20191
(703) 716-1191

日本国特許庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出願年月日 2002年 9月27日
Date of Application:

出願番号 特願2002-284164
Application Number:

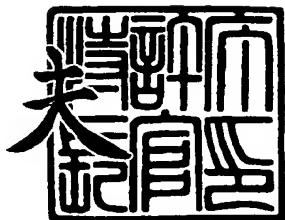
[ST. 10/C] : [JP 2002-284164]

出願人 松下電器産業株式会社
Applicant(s):

2003年 9月 1日

特許庁長官
Commissioner,
Japan Patent Office

今井康



【書類名】 特許願

【整理番号】 2030744027

【提出日】 平成14年 9月27日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 15/16

【発明者】

【住所又は居所】 大阪府門真市大字門真1006番地 松下電器産業株式会社内

【氏名】 高木 利匡

【発明者】

【住所又は居所】 大阪府門真市大字門真1006番地 松下電器産業株式会社内

【氏名】 前田 茂則

【発明者】

【住所又は居所】 ドイツ国 ランゲン 63225 モンツァシュトラーセ 4シ- パナソニック ヨーロピアン ラボラトリーズ ゲーエムベーハー内

【氏名】 ヨーク フォグラー

【発明者】

【住所又は居所】 ドイツ国 ランゲン 63225 モンツァシュトラーセ 4シ- パナソニック ヨーロピアン ラボラトリーズ ゲーエムベーハー内

【氏名】 ゲラルド ファイファー

【特許出願人】

【識別番号】 000005821

【氏名又は名称】 松下電器産業株式会社

【代理人】

【識別番号】 100105050

【弁理士】

【氏名又は名称】 鷲田 公一

【手数料の表示】

【予納台帳番号】 041243

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9700376

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 データ処理装置、データ処理方法及びデータ処理プログラム

【特許請求の範囲】

【請求項 1】 1 以上のコンポーネントを保持するデータベースを有しており、前記コンポーネントを読み込んだファイルに適用して処理を施すデータ処理装置であって、前記データベース内には前記ファイルのファイル情報に基づいて処理に適したコンポーネントを判定するタイプ判定関数を有することを特徴とするデータ処理装置。

【請求項 2】 前記タイプ判定関数は、前記コンポーネント各々の内部に保持されていることを特徴とする請求項 1 記載のデータ処理装置。

【請求項 3】 前記データベースは、前記タイプ判定関数の参照先を示す判定関数エントリを保持することを特徴とする請求項 1 または請求項 2 記載のデータ処理装置。

【請求項 4】 前記判定関数エントリは、各コンポーネントと 1 対 1 に対応して保持されることを特徴とする請求項 3 記載のデータ処理装置。

【請求項 5】 前記データベースに保持されるコンポーネントの中から処理に適したコンポーネントを検索するための判定手段をさらに有し、

前記判定手段は、前記判定関数エントリを前記データベースから読み込み、さらに前記判定関数エントリが参照先として示す前記タイプ判定関数を前記データベースより読み込み、処理対象であるファイルのファイル情報を前記判定関数に適用して判定することより、処理に適したコンポーネントを検索することを特徴とする請求項 4 記載のデータ処理装置。

【請求項 6】 前記判定手段により検索されたコンポーネントを読み込んで組合せることにより、目的とする処理を実行する処理部をさらに備え、

前記処理部内に存する制御手段は、コンポーネントの出力ファイルのファイル情報を読み込んで前記判定手段に伝えることにより、前記出力ファイルに対する次の処理に適したコンポーネントを選択することを可能とすることを特徴とする請求項 5 記載のデータ処理装置。

【請求項7】 対象とするデータの処理に適したコンポーネントであるか否かを判定するタイプ判定関数と前記タイプ判定関数を参照先として出力する判定関数エントリとを保持するデータベースを備えたデータ処理装置のデータ処理方法であって、

前記判定関数エントリを前記データベースから読み込む第1のステップと、

前記判定関数エントリが参照先として示す前記タイプ判定関数を前記データベースより読み込む第2のステップと、

前記タイプ判定関数に処理対象であるデータの情報を適用し、適用可否の結果を得る第3のステップと、

を有し、処理に適したコンポーネントが選択されるまで前記第1のステップから第3のステップを繰り返すことを特徴とするデータ処理方法。

【請求項8】 対象とするデータの処理に適したコンポーネントをデータベースから選択するためのデータ処理プログラムであって、コンピュータに、

対象とするデータの処理に適しているコンポーネントか否かを判定するタイプ判定関数の参照先を出力する判定関数エントリを読み込むステップと、

前記判定関数エントリが参照先として示すタイプ判定関数を読み込むステップと、

前記タイプ判定関数に処理対象であるデータの情報を適用し、適用可否の結果を得るステップと、

を実行させるためのデータ処理プログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、ある機能を実現するために組み合わされたソフトウェア部品から構成されるコンポーネントを読み込んで処理を実行するデータ処理装置、およびコンポーネントデータベースに関し、特にメディアデータ処理コンポーネントの保持方法および高速な読み出し方法に関する。

【0002】

【従来の技術】

従来のデータ処理装置は、データベースにコンポーネントの名前および属性が保持されており、データ処理制御部は目的のコンポーネントを名前や属性より探し出し、また名前や属性の情報では不足している場合、実際にコンポーネント全体を読み出すように構成されているものがある（例えば、特許文献1の図1参照）。

【0003】

特許文献1に記載されているデータ処理装置（文献中では、資源管理装置と呼ばれている）の全体構成を図10に示す。

【0004】

同図に示すデータ処理装置800は、コンテキスト保持部801-1～N、属性保持部802-1～N、名前解析部803-1～N、名前変換部804-1～N、名前解決部805、資源実現部806、資源保持部807-1～M、資源管理部808、通信部809、コンテキストデータベース部810、コンテキストデータベース検索部811、コンテキスト選択部812、及びコンテキスト抽出部813から構成されている。

【0005】

コンテキスト保持部801-1～801-Nには、属性保持部802-1～802-N、名前解析部803-1～803-N、名前変換部804-1～804-Nがそれぞれ設けられている。属性保持部802-1～802-Nには、それぞれコンテキスト保持部801-1～801-Nに付けられた任意の個数の属性が、その属性名と属性値のペアのリストとして保持されている。

【0006】

名前解析部803-1～803-Nは、それぞれ入力される資源の名前を解析する。名前変換部804-1～804-Nは、それぞれ名前解析部803-1～803-Nで解析された結果を受け、解析結果を資源実現表現に変換する。この資源実現表現は、資源検索式を要素とする資源要素列と、その資源要素列を処理するための手続き情報との組から構成される。ここで、資源検索式は、計算機システム内にあらかじめ実現されている実資源を指定するための検索式である。

【0007】

名前解決部805は、コンテキスト保持部801-1～Nが出力した資源実現表現を受け取り、その資源実現表現のすべてあるいはいずれかに対応する名前解決結果が資源実現部806に対して出力される。

【0008】

資源実現部806は、名前解決部805から出力された名前解決結果を入力し、その名前解決結果に対応する資源のハンドルを出力する。

【0009】

資源保持部807-1～Mは、それぞれあらかじめ実現されている資源を保持するとともに、その資源を識別するための識別子を保持している。

【0010】

資源管理部808は、資源保持部807-1～Mが保持している資源を管理するとともに、前記資源検索式に従って資源を特定する。

【0011】

通信部809は、名前解決部805、資源実現部806、資源管理部808、及びコンテキストデータベース部810間の通信を行う。

【0012】

コンテキストデータベース部810は、N個のコンテキスト保持部801-1～Nをコンテキスト集合として管理しており、コンテキストデータベース検索部811、コンテキスト選択部812、コンテキスト抽出部813を有している。

【0013】

コンテキストデータベース検索部811は、属性を用いて記述されているコンテキスト検索式を受け取り、そのコンテキスト検索式を満足する属性を有しているコンテキスト保持部801-i（iは1～Nの整数）をコンテキスト集合から探し出す。検索結果は、例えばコンテキスト保持部801-iに対するリファレンスの集合として出力することができる。

【0014】

コンテキスト選択部812は、コンテキストデータベース間等の通信を行う通信部809にコンテキスト集合の抽出を行うためのコンテキスト検索式が入力されたとき、このコンテキスト検索式をコンテキストデータベース検索部811に

渡して、コンテキスト検索式を満足するコンテキスト保持部801-iの集合を得て、コンテキスト抽出部813に出力する。

【0015】

コンテキスト抽出部813は、コンテキスト保持部801-iに対するリファレンスの集合を入力とし、そのリファレンスによって示されるコンテキスト保持部801-iと等価なコンテキスト保持部を合成するために必要な情報をコンテキスト表現として出力する。

【0016】

例えば、データ処理装置800において、属性保持部802-1～Nには、所有者を表わす属性“owner”、変更日時を表わす“mdate”、及びカテゴリを表わす“category”的3つの属性が、属性名と属性値のペアのリストとして保持されているものとする。ここでは、属性保持部802-1の“owner”的値は“Smith”、“mdate”的値は95.3.18、及び“category”的値は“picture”であり、属性保持部802-2の“owner”的値は“Smith”、“mdate”的値は95.11.14、及び“category”的値は“picture”であり、属性保持部802-3～Nの全ての“owner”的値は“Bill”、“mdate”的値は95.3.3、及び“category”的値は“sound”であるとして、コンテキストの抽出を行う動作を中心に説明する。

【0017】

まず、例えば所有者を表わす属性“owner”的値が“Smith”に等しく、かつ、変更日時を表わす属性“mdate”的値が95.3.16よりも大きく、かつ、カテゴリ属性“category”的値が“picture”に等しいコンテキスト保持部801を検索する場合を考える。

【0018】

コンテキスト選択部812は、コンテキストデータベース検索部811に上記検索を実現するための検索式を渡し、この検索式を満足する属性を属性保持部802に有するコンテキスト保持部801を検索する。検索式は例えば、C++ ObjectQueryLanguageに準拠した場合は図11に示すような

記載となる。この検索式をコンテキスト保持部 801-1 およびコンテキスト保持部 801-2 のみが満たすので、検索結果としてコンテキストデータベース検索部 811 は、コンテキスト保持部 801-1 のリファレンスと、コンテキスト保持部 801-2 のリファレンスとからなる集合をコンテキスト選択部 812 に返す。

【0019】

コンテキスト保持部 801-1 のリファレンスと、コンテキスト保持部 801-2 のリファレンスとからなる集合を受け取ったコンテキスト選択部 812 は、受け取った集合をコンテキスト抽出部 813 に入力し、リファレンスに該当するコンテキスト保持部 801-1 及びコンテキスト保持部 801-2 を取り出すよう要求することにより、該当するコンテキストを取得することができる。

【0020】

【特許文献 1】

特開平 9-171501 号公報

【0021】

【発明が解決しようとする課題】

しかしながら、従来のデータ処理装置では、従来の名前や属性では識別できないような新規の属性が付加されたコンテキストが追加された場合、それに合わせて従前のコンテキスト保持部も再構築しなければならないという問題があった。

【0022】

具体的には、例えば、ファイルサイズを示す属性 “size” が追加された場合は、全てのコンテキスト保持部を更新する必要があり、コンテキスト保持部の数が多い場合や、頻繁に新しい属性が追加されている場合は、コンテキスト保持部のメンテナンスの負荷が大きくなってしまうという問題が発生する。

【0023】

本発明は、このような問題を解決するためになされたもので、新規コンポーネントが追加された場合でもデータベースの再構築を不必要とするデータ処理装置、データ処理方法及びデータ処理プログラムを提供することを目的とする。

【0024】

【課題を解決するための手段】

上記課題を解決するために本発明においては、コンポーネント内にコンポーネント本体とは別に、対象とするデータの処理に適するコンポーネントであるか否かを判定するタイプ判定関数を有し、データ処理装置には、読み込んだファイルまたは前段のコンポーネントにより処理され出力されたデータの情報を前記タイプ判定関数に適用する判定手段を備え、各コンポーネントと対応するタイプ判定を実施することで、判定のための属性が変更または追加となったとしても、タイプ判定関数のみを変更すれば足り、コンポーネント本体（従来のコンテキスト保持部全体に相当）を更新する必要はないため、効率的にデータベースを更新することが可能となる。

【0025】

【発明の実施の形態】

以下、本発明の実施の形態について図1から図9を用いて説明する。なお、本発明は本実施の形態に何ら限定されるものではなく、その要旨を逸脱しない範囲において、様々な形態で実施し得る。

【0026】

図1は、本発明の実施の形態におけるデータ処理装置100を示したブロック図である。図1において、メディアコンポーネントデータベース101は、メディアを処理するためのコンポーネントであるメディアコンポーネントを保持している。メディア判定部102は、メディアコンポーネントを比較して判別する。操作入力部103は、ユーザからの入力を受け付ける。ファイルシステム104は、マルチメディアデータを保持している。メディア処理部105は、メディアの処理を行う。映像表示部111は、映像を表示する。音声出力部112は、音声を出力する。

【0027】

図2は、メディアコンポーネントデータベース101が保持するコンポーネントの保持形態を示した図である。

【0028】

メディアコンポーネントデータベース101には、メディアコンポーネント3

0-1～Nが保持されており、各メディアコンポーネントの中にはそのコンポーネントがどのような種類のデータに適用されるべきコンポーネントであるかを判定するためのメディアタイプ判定関数40-1～Nが含まれている。また、メディアコンポーネントデータベース101には、メディアタイプ判定関数40-1～Nの各々への参照先アドレスを返す関数であるメディアコンポーネントメディアタイプ判定関数エントリ20-1～Nが保持されている。

【0029】

以上のように構成されたデータ処理装置100について、図3、図4を用いてその動作を説明する。

【0030】

図3は、データ処理装置100がマルチメディアファイルを再生する手順を示したフロー図である。

【0031】

データ処理装置100のユーザからファイルの再生要求が操作入力部103に入力され、ファイル名と再生要求がメディア処理部105内のメディア制御部106に送られる(S1010)。

【0032】

メディア制御部106は、指定されたファイルを読み込むためのコンポーネント(ファイル読み込みコンポーネント)をメディア判定部102へ要求する(S1020)。

【0033】

メディア判定部102は、指定されたファイル名をもとにファイルを読み込むためのコンポーネント(ファイル読み込みコンポーネント)をメディアコンポーネントデータベース101より検索する(S1030)。

【0034】

この検索を行う際は、図4に示すフロー図に従った処理を行う。

【0035】

すなわち、まず、メディア判定部102は、メディアコンポーネントデータベース101よりメディアタイプ判定関数エントリ20-1を読み込む(S201

0）。なお、図4において、xの初期値は1であるものとする。次に、読み出したメディアタイプ判定関数エントリ20-1からの参照先であるメディアコンポーネント30-1のメディアタイプ判定関数40-1と、ユーザが操作入力部103を介して指定したファイル名に対するファイル名とを用いて、このメディアコンポーネント30-1が要求されたファイルの読み込みに適しているか否かが判定される（S2020）。

【0036】

メディアタイプ判定関数40-1の判定の結果、メディアコンポーネント30-1が、ユーザが指定したファイル名に対応するデータの処理コンポーネントとして適していなかった場合は、xの値を1だけ増加させ（S2040）、次のメディアタイプ判定関数エントリ20-2を読み込む（S2010）。一方、処理コンポーネントとして適していた場合は、メディア判定部102は、メディアコンポーネントデータベース101よりメディアコンポーネント30-1を読み出し、メディア制御部106へ返す（S2030）。

【0037】

メディア判定部102にてこのような処理を行うことによって、メディアコンポーネント全体を読み出すことなく、ユーザ指定のデータの処理コンポーネントとして適しているか否かの判定を行うことができ、メディアコンポーネントの検索の高速化を図ることができる。

【0038】

また、メディア判定のための方法が追加や変更された場合でも、コンポーネント内のメディアタイプ判定関数40-1～Nのみの変更で済み、データベース自体の再構築が不要となる。すなわち、例えば、新しいタイプのメディアデータを処理するコンポーネントが追加された場合、従来の技術ではコンテキスト保持部を再構築することが必要となるが、本願発明のデータ処理装置は、各コンポーネント本体は再構築する必要がなく、メディアタイプ判定関数40-1～Nのみを変更することによって、新しいタイプのメディアデータの処理が可能であるか否かを判別することが可能となる。

【0039】

図3のフロー図を参照して、メディア制御部106は、読み込んだファイル読み込コンポーネント107を用い、ファイルシステム104よりファイルを読み込む（S1040）。

【0040】

メディア制御部106は、読み込んだファイルの情報から多重化情報を取得し、その逆多重化を行うためのコンポーネント（逆多重化コンポーネント）をメディア判定部102へ要求する（S1050）。

【0041】

なお、多重化情報とは、ファイルシステム104に保持されているファイルに付加されている情報であって、そのファイルに多重化されている複数のデータタイプが記載されている情報である。以下、多重化情報について具体的に説明する。

【0042】

図5は、多重化されたファイルの状態を概念的に示した図である。多重化ファイル（MP4 file）には、大きく分けてファイルの書誌的な情報を記載している領域（Movie Atom）と、メディアの実体を格納している領域（Media Atom）とに分かれる。

【0043】

なお、“Movie Atom”領域の構造は、いわゆる入れ子構造となっており、“Movie Atom”領域の中は、さらに“Movie Header Atom”と“Track Atom”といった複数の領域に区分けされて情報が保持されている。

【0044】

メディア制御部106は、上記“Movie Atom”の中からメディアタイプ判定関数40-1～Nの引数として必要な書誌的情報を、上記多重化ファイルの記載箇所を特定した情報として取得する。この情報を多重化情報と呼ぶ。

【0045】

図6は、メディア制御部106が取得する多重化情報の例を示したものである。図6において、例えば「メディアタイプ」は、多重化ファイル中の“Movie”

e Atom”の中の“Track Atom”の中の“Media Atom”の中の“Handler Reference”に記載されていることを示している。

【0046】

なお、ここでは指定されたファイルが、メディアデータを多重化しているマルチメディアデータであることにするが、本発明は、このデータに限定されるものではない。

【0047】

再度、図3のフロー図を参照して、メディア判定部102は、メディア制御部106から入力された多重化情報をもとに、逆多重化を行うためのコンポーネント（逆多重化コンポーネント）をメディアコンポーネントデータベース101から検索する（S1050）。ここでもS1030での処理と同様に図4の処理を行い、対象とするデータを逆多重化するのに適したコンポーネントを選択する。

【0048】

メディア制御部106は、読み込んだ逆多重化コンポーネント108を用い、ファイル読込コンポーネント107からマルチメディアデータを読み込み、逆多重化を行う（S1070）。

【0049】

メディア制御部106は、逆多重化されたメディアデータに付随するメディア情報（映像）を取得し、その映像を表示するためのコンポーネント（映像出力コンポーネント）をメディア判定部102へ要求する（S1080）。

【0050】

ここでいうメディア情報は、具体的には例えば図7に示すような形式のものである。メディア情報には、図7に示すようにメディアタイプ、メディアサブタイプ等のアイテムとそれに対応する値や文字列とが記載されている。図7において、“None”は該当するアイテムについて記載がないことを示している。

【0051】

メディア判定部102は、メディア制御部106から入力されたメディア情報をもとに、映像を表示するためのコンポーネント（映像出力コンポーネント）を

メディアコンポーネントデータベース101から検索する（S1090）。ここでもS1030での処理と同様に図4の処理を行い、対象とするデータを映像出力するのに適したコンポーネントを選択する。

【0052】

この場合のメディアタイプ判定関数40-1～Nの動作を示すフロー図を図8に示す。

【0053】

まず、メディアタイプ判定関数40-1～Nは、上流のコンポーネントの出力から図7に示すメディア情報を取得する（S3010）。ここで、上流のコンポーネントとは、1つ前の処理が行われたコンポーネントを指す。したがって、ここでは、逆多重化コンポーネント108が上流のコンポーネントに該当する。処理は、メディア判定関数40-1～Nに対して適用され、適したコンポーネントが見つかるまで繰り返される。以下、メディアタイプ判定関数40-1についての動作について説明する。

【0054】

メディアタイプ判定関数40-1は、読み込んだメディア情報のアイテムと、予め保持しているテーブルとを比較し、各アイテムがメディア判定関数の保持するテーブルの範囲内にあるか否かを判定する（S3020）。

【0055】

図9は、メディアタイプ判定関数40-1が保持しているテーブルの一例を示したものである。メディア判定関数40-1は、各アイテムについて、順番に比較していくが、メディアタイプ判定関数40-1の保持しているテーブルの値の範囲内に無い場合は、その時点で判定は“NG”となる（S3050）。逆に範囲内に入っている場合は、まだメディア判定関数40-1が判定していないアイテムがあるか否かを判断する（S3030）。まだ全てのアイテムを判断していない場合は、S3020に戻る。全てのアイテムについて判定した場合は、判定は“OK”となる（S3040）。

【0056】

なお、図9中の幾つかのアイテムについては“unspecified”（不

特定の任意の値で良い) となっているが、このようなアイテムについては比較する必要がないので、範囲内に入っているか否かの判定を行わないようにすることもできる。

【0057】

以上がメディアタイプ判定関数40-1の動作である。同様の動作を各メディア判定関数40-2～Nが行うことにより、処理に適したコンポーネントを選択することができる。

【0058】

図3のフロー図を参照して、メディア制御部106は、読み込んだ映像出力コンポーネント109の出力先を映像表示部111へ接続する(S1100)。

【0059】

そして、メディア制御部106は、逆多重化されたメディアデータのメディア情報(音声)を取得し、その音声を出力するためのコンポーネント(音声出力コンポーネント)をメディア判定部102へ要求する(S1110)。

【0060】

メディア判定部102は、メディア制御部106から入力されたメディア情報をもとに、音声を出力するためのコンポーネント(音声出力コンポーネント)をメディアコンポーネントデータベース101から検索する(S1120)。ここでもS1030での処理と同様に図4の処理を行い、対象とするデータを音声出力するのに適したコンポーネントを選択する。

【0061】

メディア制御部106は、読み込んだ音声出力コンポーネント110の出力先を音声出力部112へ接続する(S1130)。

【0062】

メディア制御部106は、各コンポーネントへ再生開始を要求する(S1140)。

【0063】

映像表示部111、音声出力部112において、映像、音声が出力される(S1150)。

【0064】

以上のように、メディアコンポーネントを検索する際、メディアタイプ関数40-1～Nを用いることによって、メディアコンポーネント全体を読み出すことなく、そのコンポーネントが処理に適しているか否かの判定を行うことができ、メディアコンポーネントの検索の高速化を図ることができる。

【0065】

また、メディアの判定のための方法が追加や変更された場合でも、コンポーネント内のメディアタイプ判定関数40-1～Nのみの変更で済み、データベース自体の再構築が不要となる。

【0066】

【発明の効果】

以上のように本発明によれば、コンポーネント内にコンポーネント本体とは別に、対象とするデータの処理に適するコンポーネントであるか否かを判定するタイプ判定関数を有し、データ処理装置には、読み込んだファイルまたは前段のコンポーネントにより処理され出力されたデータの情報を前記タイプ判定関数に適用する判定手段を備え、各コンポーネントと対応するタイプ判定を実施することで、判定のための属性が変更または追加となったとしても、タイプ判定関数のみを変更すれば足り、コンポーネント本体（従来のコンテキスト保持部全体に相当）を更新する必要はないため、効率的にデータベースを更新することが可能となる。

【0067】

また、メディアタイプ判定関数エントリを利用することによって、データベースよりコンポーネント全てを取得することなく、メディアタイプ判定関数エントリの参照先であるタイプ判定関数のみを読み込めばコンポーネントの評価を行うことができ、目的のコンポーネントを取得するまでの時間を短縮することができる。

【図面の簡単な説明】

【図1】

本発明の実施の形態におけるデータ処理装置の構成を示すブロック図

【図2】

本発明の実施の形態におけるメディアコンポーネントデータベースの構成を示す図

【図3】

本発明の実施の形態におけるマルチメディアファイル再生手順を示すフロー図

【図4】

本発明の実施の形態におけるメディア判定部の処理手順を示すフロー図

【図5】

本発明の実施の形態における多重化ファイルの構成を示す図

【図6】

本発明の実施の形態における多重化情報の一例を示す図

【図7】

本発明の実施の形態におけるメディア情報の一例を示す図

【図8】

本発明の実施の形態におけるメディア判定関数の動作を示すフロー図

【図9】

本発明の実施の形態におけるメディアタイプ判定関数が保持するテーブルの一例を示す図

【図10】

従来のデータ処理装置の構成を示すブロック図

【図11】

従来のデータ処理装置における検索式の一例を示す図

【符号の説明】

101 メディアコンポーネントデータベース

102 メディア判定部

103 操作入力部

104 ファイルシステム

105 メディア処理部

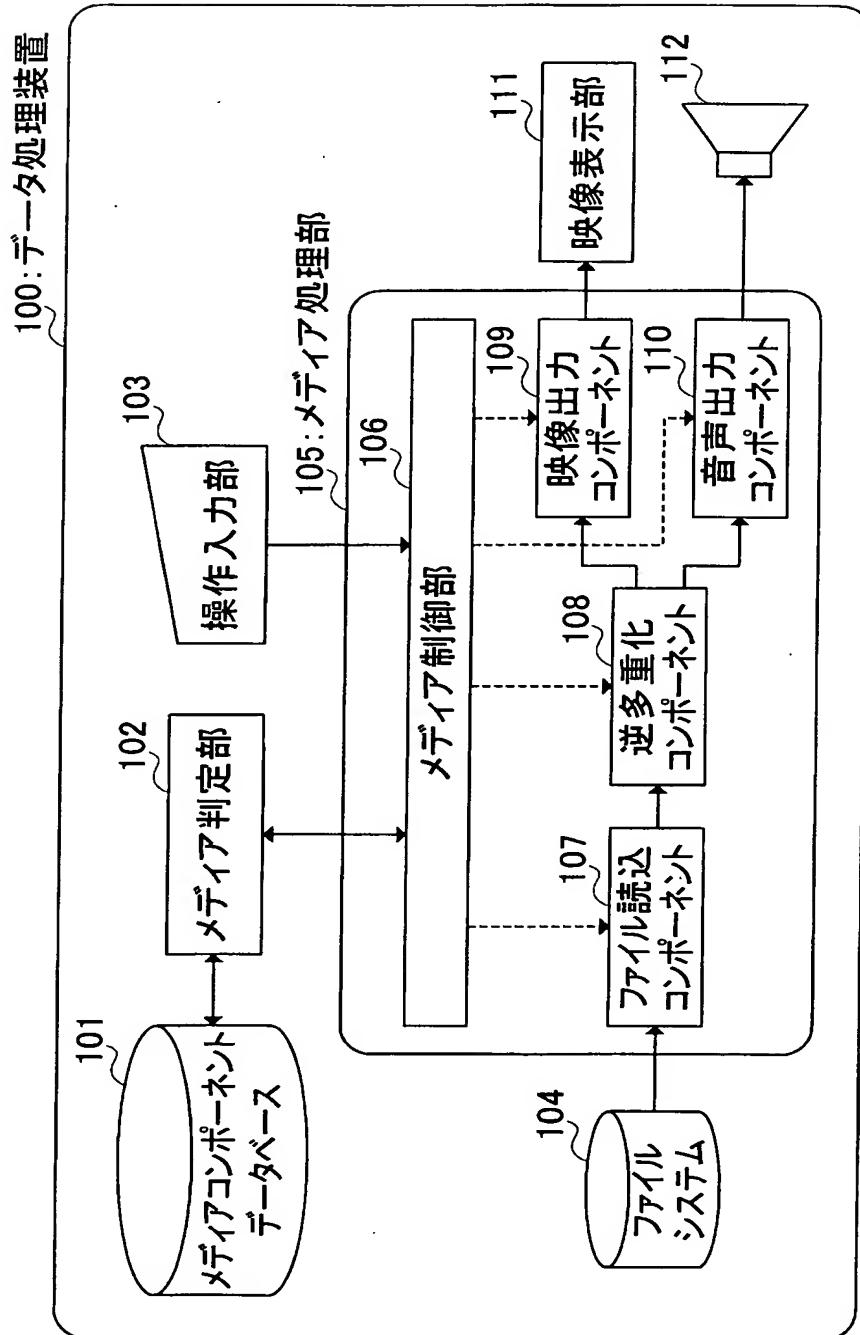
106 メディア制御部

- 107 ファイル読込コンポーネント
- 108 逆多重化コンポーネント
- 109 映像出力コンポーネント
- 110 音声出力コンポーネント
- 111 映像表示部
- 112 音声出力部
- 20-1～N メディアタイプ判定関数エントリ
- 30-1～N メディアコンポーネント
- 40-1～N メディアタイプ判定関数

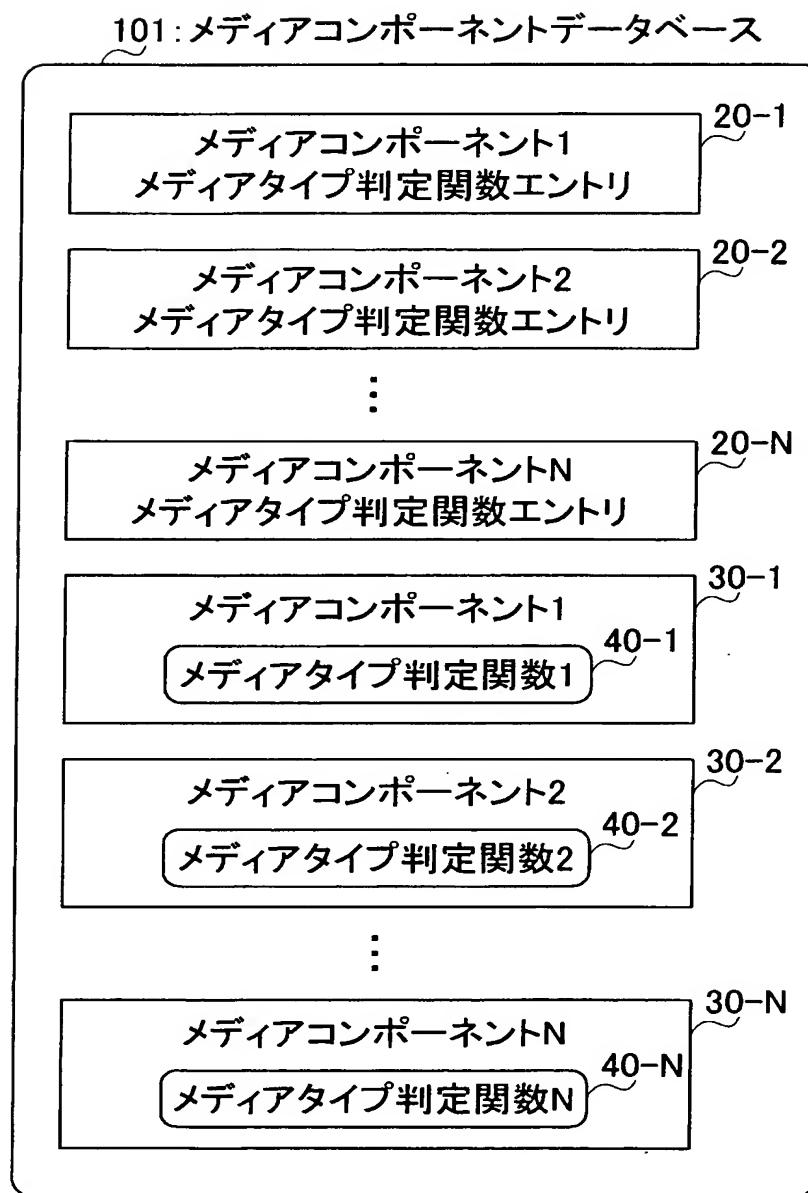
【書類名】

図面

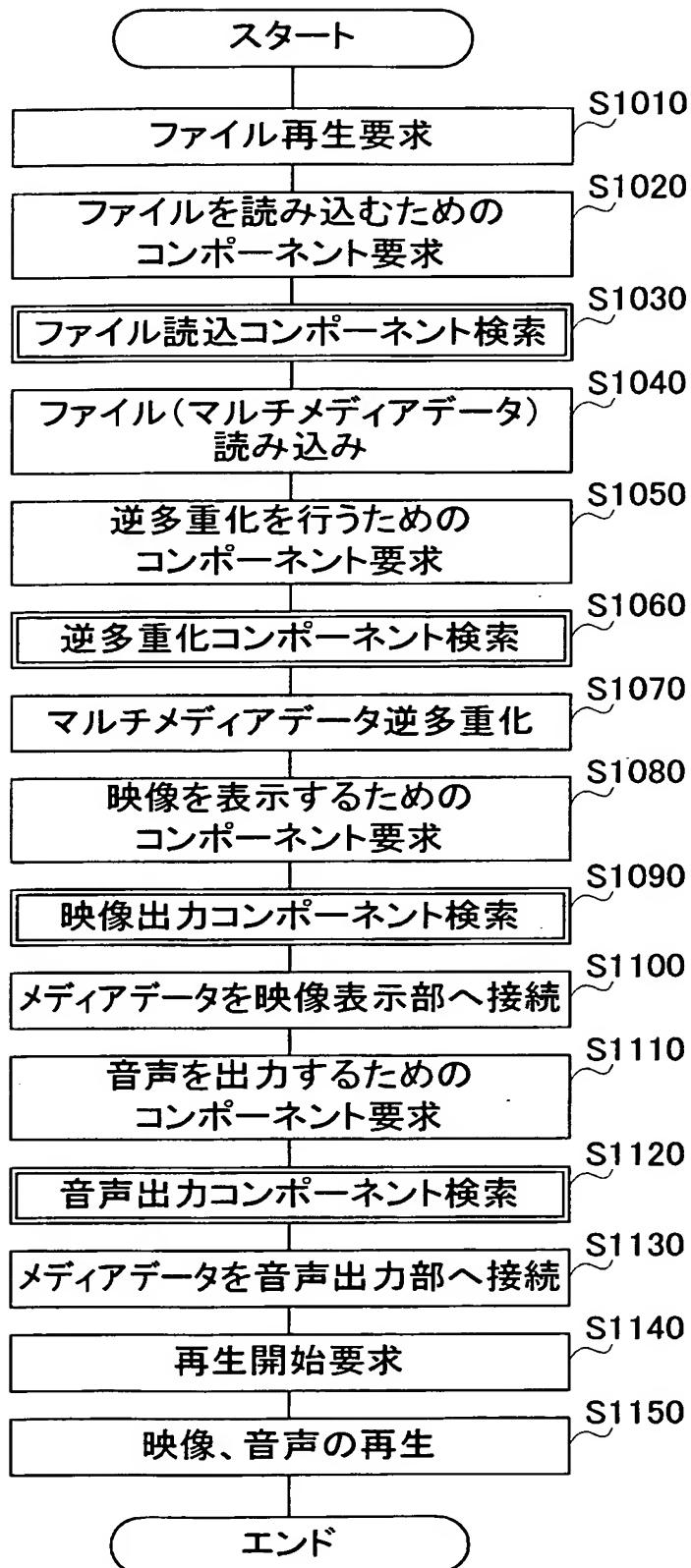
【図 1】



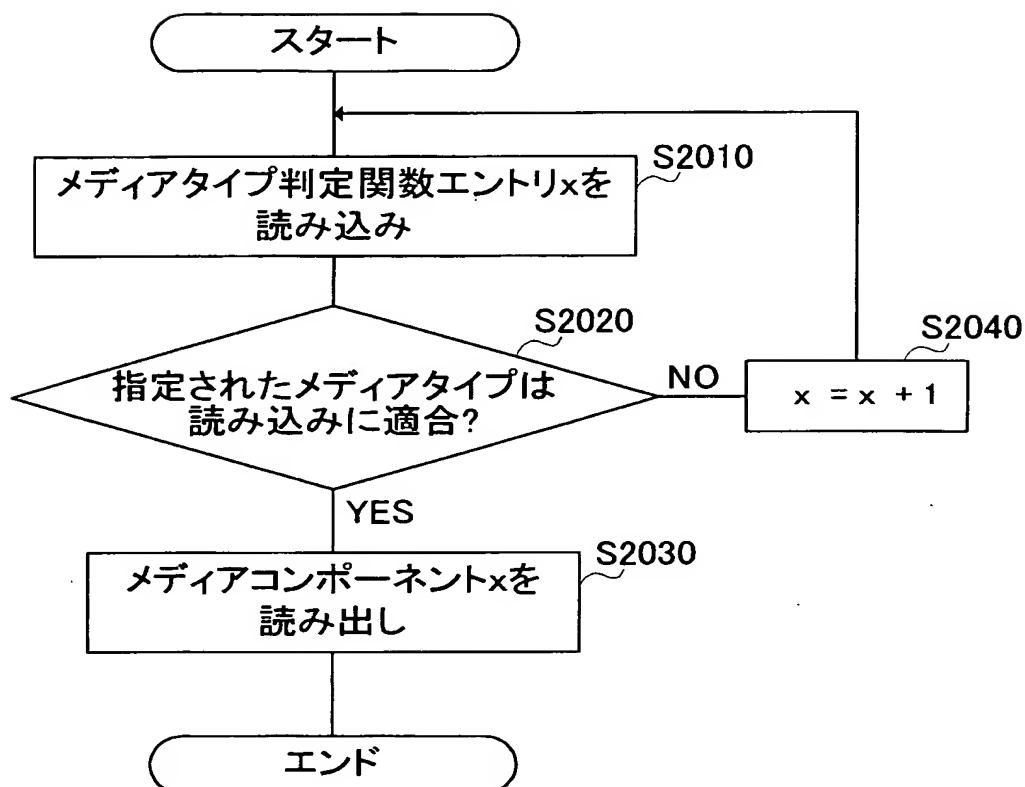
【図2】



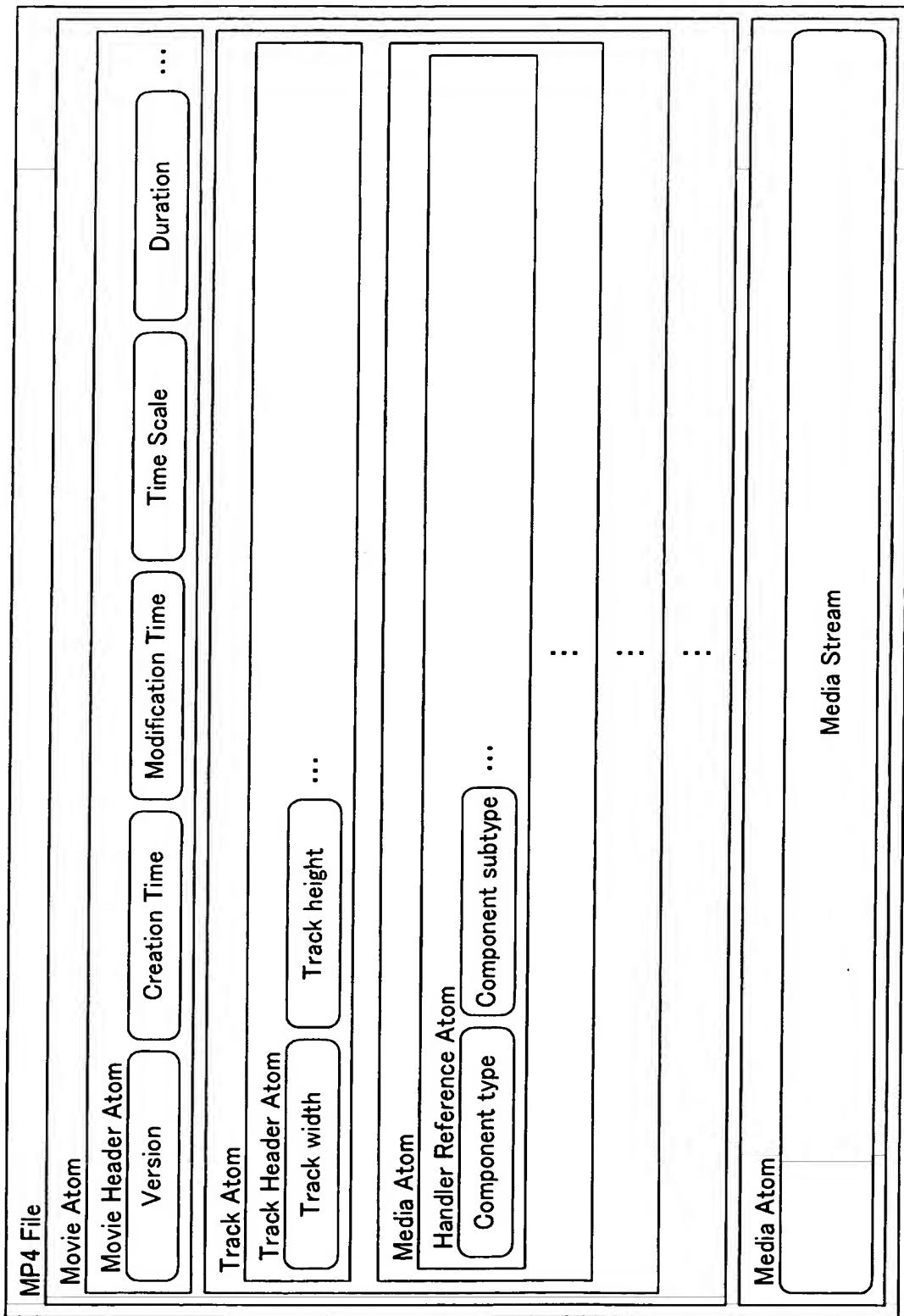
【図3】



【図4】



【図5】



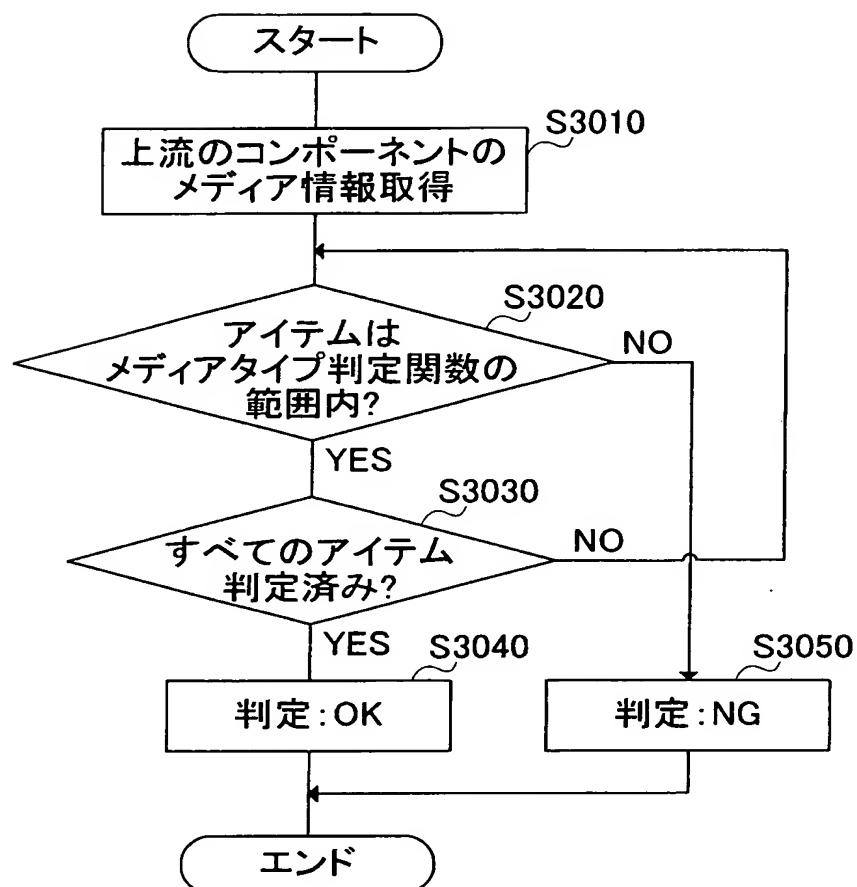
【図 6】

	MP4 Atom
メディアタイプ	Movie->Track->Media->Handler Reference(Component subtype)
メディアサブタイプ	Movie->Track->Media->Media Information->Sample Table ->Sample Description->Visual Sample Entry(Type)
コーデック	Movie->Track->Media->Media Information->Sample Table ->Sample Description->Visual Sample Entry(ESD)
バージョン	Movie->Track->Media->Media Information->Sample Table ->Sample Description->Visual Sample Entry(Decoder_version)
平均ビットレート	Movie->Track->Media->Media Information->Sample Table ->Sample Description->Visual Sample Entry(ESD)
最大ビットレート	Movie->Track->Media->Media Information->Sample Table ->Sample Description->Visual Sample Entry(ESD)
チャネル数	None
サンプリング周波数	None
幅	Movie->Track->Media->Media Information->Sample Table ->Sample Description->Visual Sample Entry(Width)
高さ	Movie->Track->Media->Media Information->Sample Table ->Sample Description->Visual Sample Entry(Height)

【図7】

アイテム	逆多重化後
メディアタイプ	Video
メディアサブタイプ	MPEG-4
コーデック	Simple Profile
バージョン	1.0
平均ビットレート	400
最大ビットレート	600
チャネル数	None
サンプリング周波数	None
幅	176
高さ	144

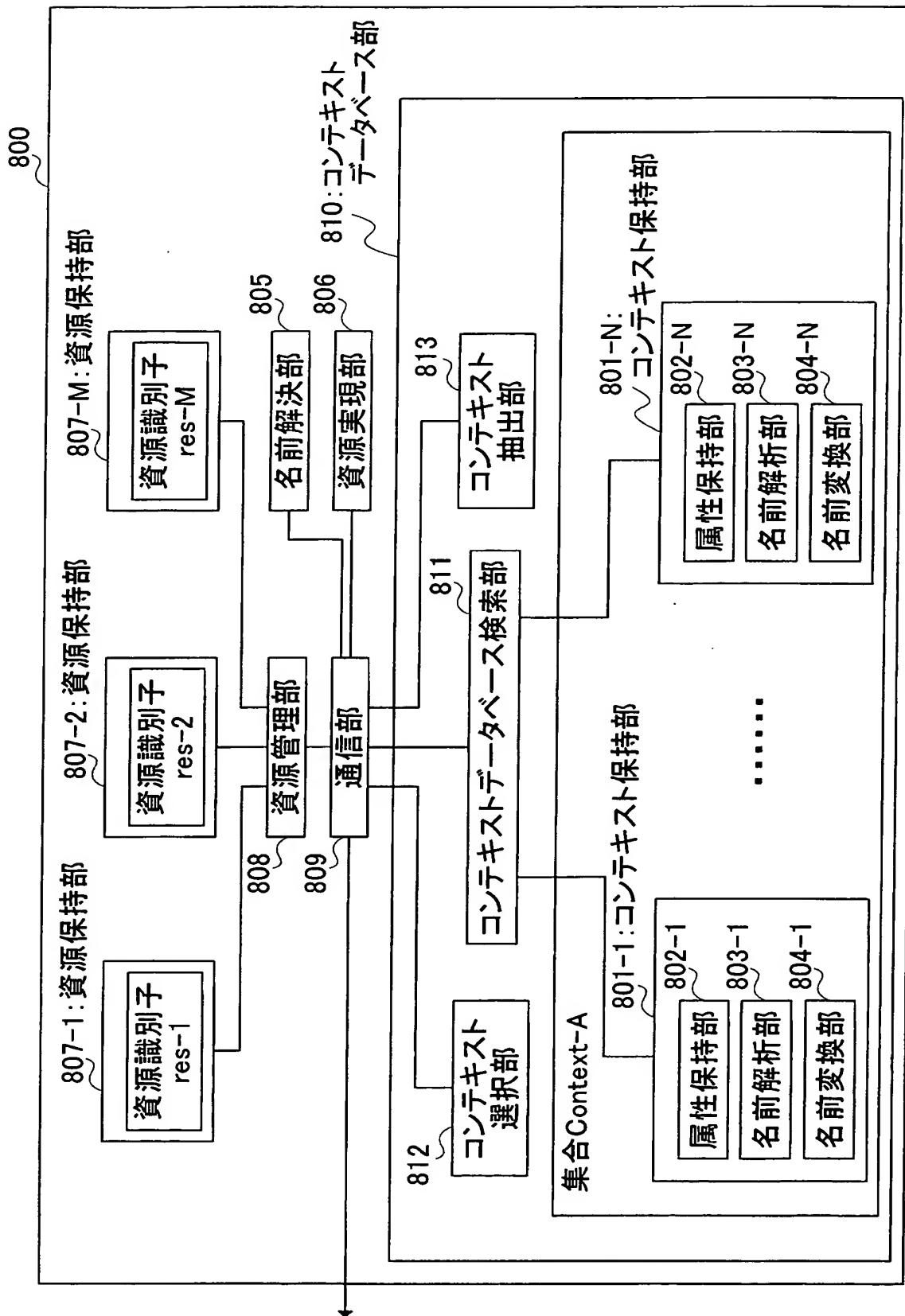
【図8】



【図9】

アイテム	メディアタイプ判定関数
メディアタイプ	Stream
メディアサブタイプ	MP4
コーデック	unspecified
バージョン	<=1.0
平均ビットレート	unspecified
最大ビットレート	unspecified
チャネル数	unspecified
サンプリング周波数	unspecified
幅	unspecified
高さ	unspecified

【図 1 0】



【図11】

```
Set<Ref<Class Context>>subcontext:  
Context->query(subcontext.  
"select distinct x from Context-A where \  
x.property.owner='Smith' \  
and x.property.ndata>95.3.16 \  
and x.property.category='picture'");
```

【書類名】 要約書

【要約】

【課題】 データベースからコンポーネント化されたマルチメディアデータ処理部を取り出すとき、マルチメディアデータ処理部を一旦データベースから取り出し、そのマルチメディアデータ処理部と他のマルチメディアデータ処理部との条件判定を行う必要があり、マルチメディアデータ処理部の検索に時間がかかる。

【解決手段】 データベースへマルチメディアデータ処理部本体だけではなく、マルチメディアデータ処理部の条件判定を行う関数の関数エントリも登録しておき、マルチメディアデータ処理部の検索時には、まず関数エントリのみ取り出し条件判定を行い、目的のものであった場合のみマルチメディアデータ処理部本体を取り出す。これによって、マルチメディアデータ処理部の検索の時間が短縮される。

【選択図】 図 1

特願2002-284164

出願人履歴情報

識別番号 [000005821]

1. 変更年月日 1990年 8月28日

[変更理由] 新規登録

住所 大阪府門真市大字門真1006番地
氏名 松下電器産業株式会社